UNIVERSITÉ DE MONTRÉAL

CENTRE DE RECHERCHE SUR LES TRANSPORTS

CASE POSTALE 6128, STATION A, MONTRÉAL, QUÉBEC, H3C 3J7

A METHOD FOR COMPUTING NETWORK EQUILIBRIUM WITH ELASTIC DEMANDS

by

Michael Florian and Sang Nguyen

This research was supported by a grant from the Ford Motor Company of Canada for applied research in the Analysis and Control of Urban Transportation Network.

Université de Montréal

Publication # 140 - Département d'informatique
Publication # 2 - Centre de recherche sur les transports

Août 1973
Revisé Janvier 1974

ABSTRACT

We present a new method for computing the equilibrium flows in a transportation network with elastic demands and congestion effects for a single class of users. The method that we shall develop in the following results from the combination of a fixed demand assignment method and an application of Generalized Benders Decomposition. We give computational results obtained with the algorithm.

RESUME

On présente une nouvelle méthode pour calculer les flots d'équilibre dans un réseau de transport avec demande élastique et coûts de congestion pour une seule classe d'usagers. La méthode qu'on développe est basée sur la méthode de décomposition généralisée de Benders. On présente aussi des résultats numériques obtenus avec cet algorithme.

1. Introduction

The prediction of flows in a transportation network is equivalent, in theory [1] , to finding the flows that establish the equilibrium between a supply function S , determined by the travel time characteristics of the links of the networks and a demand function D , determined by the perception of users of their benefit for travel. The determination of these equilibrium flows is a complex task, since flows by several modes, for several classes of users, from many origins to many destinations interact in their use of the available service capacity. Although the problem is difficult to formulate mathematically in its generality, some less general versions have been formulated as nonlinear optimization problems in a transportation network. Dafermos and Sparrow [3] and Dafermos [4] consider the problem of finding equilibrium flows for a single class of users, when the demand for travel between the origin and destination zones are fixed and develop an algorithm to solve this problem, also commonly known as capacity restrained assignment. In a later paper, Dafermos [5] extends the algorithm to multiple classes of users with fixed travel demands, a problem which was also considered by Netter [14]. Computational experience with moderate size problems is not reported in either [3] or [4]. Some recent work by Sang Nguyen [15] reports on computational experience with efficient algorithms that have been developed and coded for the fixed demand

problem for a single class of users. Leventhal et al [10] report on a particular implementation of the algorithms given in [4].

In this paper we introduce a new method for computing equilibrium flows in a transportation network with elastic demands for a single class of users. This problem has also been studied by Martin and Manheim [12], and Gibert [8], Murchland [13], Wilkie and Stefanek [20] and Wigan [18], [19]. While these represent important contributions in the study of this problem, the resulting algorithms have not been spectacularly efficient. The algorithm of Martin and Manheim [12] consists in incrementally loading the network along current shortest routes until the equilibrium conditions are satisfied. The drawback of this approach is that computation of equilibrium flows requires loading the network in small increments and consequently the calculation of prohibitively large number of shortest routes. The method is suitable however for computing approximate equilibrium. Gibert [8], also in [2], suggested an interesting algorithm for solving the problem with elastic demands. His method requires, at each iteration, the calculation of shortest and longest paths between origins and destinations. The latter is a rather difficult task which is often impeded by the presence of positive loops in the network. Although Gibert does not give any computational results, we venture to judge that this algorithm would be impeded by the longest path calculations. Murchland [13] expanded on Gibert's work, and gave general principles for developing operational algorithms for the problem. However no computational results are given. Wilkie and Stefanek [20] present a control-theoretic

approach to solve this problem, however present computational results for very small problems. A survey of the literature and possible approaches to the problem is given by Ruiter [16].

The method that we shall develop in the following has the very attractive feature that it is based on the iterative use of an algorithm that computes equilibrium flows with fixed demands. Thus it is relatively simple to implement using an efficient code for the fixed demand problem. Our approach is an adaptation of Generalized Benders Decomposition [7] for this problem. We develop the new algorithm and test its efficiency using data for a medium size urban network.

2. Formulation of the Problem as a Mathematical Program

For ease of presentation we shall use the arc-chain formulation of the problem. As usual we consider a network $[N,A]$ of nodes and directed arcs, where certain nodes are identified as origins and destinations and two way streets are represented by two different direct arcs. There is, in general, a set $K(i)$ of distinct paths available for travel between the origin-destination (O-D) pair i and h_k, $k \in K(i)$ is the flow on the path k. There are I O-D pairs. We shall denote by $f_a(v_a)$ the cost of travel on the arc "a" as a function of the volume "v_a" on the arc (the supply functions). This function is assumed monotone increasing. See Figure 1. Let g_i represent the number of trips related to O-D pair i. Then the dependence of the number of trips g_i, on the cost of travel u_i between the O-D pair i is expressed by the demand function $d_i(u_i)$. This function is assumed monotone decreasing. See Figure 2. It is also assumed that the inverse function of the demand function exits, and we denote it by $W_i(g_i)$. Wigan [18] gives a good discussion of these functions. Then, the user optimized equilibrium flows $\bar{h}_k$, $k \in \bigcup_i K(i)$, $\bar{g}_i$, $i = 1,\ldots,I$ are those that maximize the consumer's surplus or equivalently, minimize.

$$\sum_{a\in A} \int_0^{v_a} f_a(v)\, dv \;-\; \sum_{i=1}^{I} \int_0^{g_i} W_i(y)\, dy \qquad (1)$$

subject to

$$\sum_{k \in K(i)} h_k = g_i \qquad i = 1,\ldots,I \qquad (2)$$

$$h_k \geq 0 \qquad k \in K(i),\ i = 1,\ldots,I \qquad (3)$$

$$g_i \geq 0 \qquad i = 1,\ldots,I \qquad (4)$$

and

$$v_a = \sum_{i=1}^{I} \sum_{k \in K(i)} \delta_{ak}\ h_k \qquad (5)$$

where

$$\delta_{ak} = \begin{cases} 1 & \text{if link } a \text{ is on path } k \\ 0 & \text{otherwise.} \end{cases} \qquad (6)$$

It is easy to see that the constraints (2) are separable for each distinct O-D pair i. The only interaction between the O-D flows occurs on the arcs and consequently in the objective function (1). Since the objective function is convex, a local minimum is also a global minimum, and the problem may be decomposed by each O-D pair i. If an efficient algorithm may be developed for a single O-D pair, then the entire problem may be solved by cyclical application of this algorithm as follows:

(1) Given initial travel demands $\bar{g}_i$, i = 1,...,I and a feasible flow pattern $\{h_k^o\}$, set p = 0, q = 0 and go to step (2).

(2) If q = I terminate, otherwise put i = p modulo (I) + 1 and go to step (3).

(3) If the current travel demand and flow $(g_i, h_k, k \in K(i))$ are in equilibrium for O-D pair i, put $q = q + 1$ and go to step (2); otherwise solve the single O-D pair i subproblem for its equilibrium $(\bar{g}_i, \bar{h}_k, k \in K(i))$, update the current link flows and costs, set $q = 1$, $p = p + 1$ and go to step (2).

Therefore, we shall single out an O-D pair i and develop our method for it. Thus, we consider the problem

$$\text{Min} \quad \sum_{a \in A} \int_o^{v_a} f_a(v)\, dv \; - \; \int_o^{g} W(y)\, dy \tag{7}$$

subject to

$$\sum_{k \in K} h_k = g \tag{8}$$

$$h_k \geq 0 \quad , \quad k \in K \tag{9}$$

$$g \geq 0 \tag{10}$$

and

$$v_a = \sum_{k \in K} \delta_{ak} h_k \; + \; C_a \tag{11}$$

where C_a is the flow on arc a resulting from the demands from all other O-D pairs.

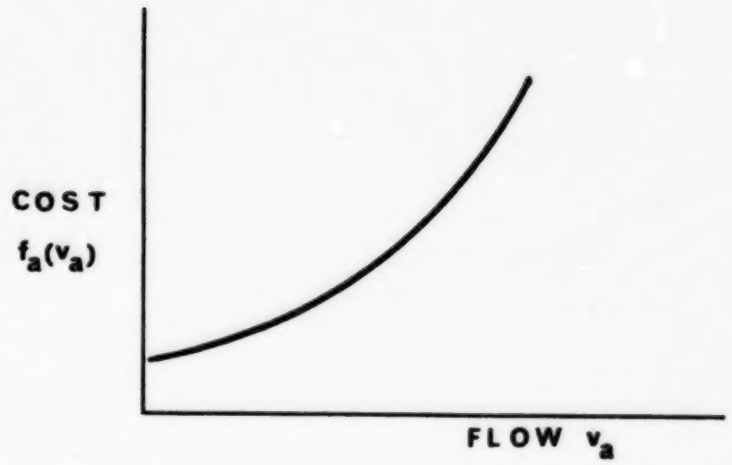

Figure 1
Supply Function

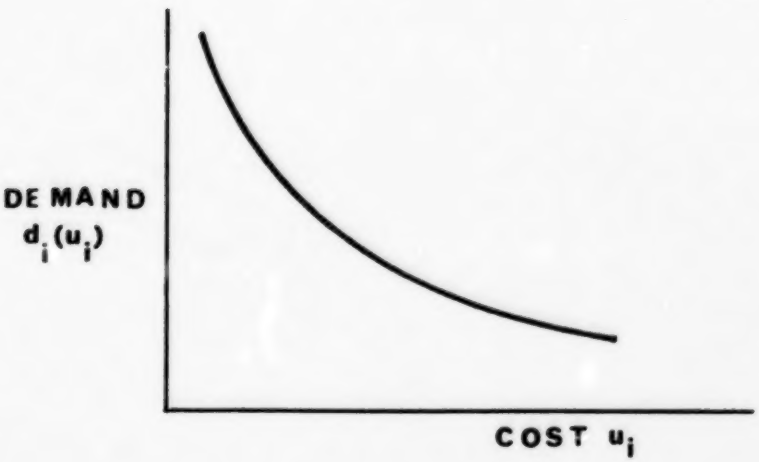

Figure 2
Demand Function

3. Development of the Algorithm

The essence of our approach is the observation that the objective function and the constraints may be partitioned into terms in h_k and terms in g. Since the objective function is convex and good algorithm are available for the fixed demand problem, the adaptation of generalized Benders Decomposition results in the following algorithm:

Step 0 Select an arbitrary g, say $\bar{g}^o$, set $x^o \leftarrow o$ and UBD $\leftarrow \infty$ and select suitable convergence parameters ϵ_1, ϵ_2. Set $J \leftarrow 1$.

Step 1 Set $\bar{g} \leftarrow \bar{g}^{J-1}$. Solve the problem -

$$\text{Min } F\{h,\bar{g}\} = \text{Min} \sum_a \int_o^{v_a} f_a(v)\, dv - \int_o^{\bar{g}} W(y)\, dy \qquad (12)$$

$$\text{subject to} \quad \sum_{k\epsilon K} h_k = \bar{g}, \qquad (13)$$

$$h_k \geq o, \quad k\epsilon K \qquad (14)$$

$$\text{where} \quad v_a = \sum_{k\epsilon K} \delta_{ak} h_k + C_a \qquad (15)$$

This is a traffic assignment problem - Let $\bar{u}^J$ be the cost of the utilized paths in the resulting assignment and $\bar{h}_k^J$ the resulting path flows.

Set UBD $\leftarrow$ min {UBD, F{ $\bar{h}_k^J$, $\bar{g}$}}

If $\bar{x}^{J-1} \geq$ UBD - ϵ_1, or $\bar{u}^J = W(\bar{g}^J) \pm \epsilon_2$, terminate

otherwise continue to Step 2.

Step 2 Solve the problem:

$$\text{Min } x^J \tag{16}$$

$$\text{Subject to } x^J \geq M^j - \int_0^g W(y)\,dy + \bar{u}^j g \,, \quad j = 1,2,\ldots,J \tag{17}$$

$$g \geq 0 \,, \quad x^J \text{ unrestricted} \tag{18}$$

and obtain $\underline{x}^J$, $\underline{g}^J$.

Set $J \leftarrow J+1$ and go to Step 1

UBD and x^J are respectively an upper bound and a lower bound of the objective function of problems (7) - (11). A detailed derivation of the constant M^J is given in the appendix.

The only apparent difficulty in using this algorithm is the solution of the problem in Step 2. Fortunately, this problem is relatively easy to solve as we show in the next section.

4. Solving the problem of Step 2

Before outlining an algorithm to solve the problem (16)-(18) we shall first state and prove the following

Proposition: In solving the problem of Step 2, the optimal value of x^J is always obtained on the last added function $\phi_J(g)$ where we have

$$\text{Min } x^J = \text{Min} \max_{1\le j\le J} \phi_j(g) \ , \ g \ge 0 \tag{19}$$

$$\text{and} \qquad \phi_j(g) = M^j - \int_0^g W(y)dy + \bar{u}^j g \ . \tag{20}$$

Proof: As all the $\phi_j(g)$ are convex functions it is known that the function defined by

$$\max_{1\le j\le J} \phi_j(g)$$

is piecewise convex over $g \ge 0$ [11]. When $J = 1$, the proposition is trivially true. For $J > 1$, as the new constraint eliminates the current minimum, it follows that

$$\bar{x}^{J-1} < \phi_J(\bar{g}^{J-1})$$

Hence, the new optimal solution of (19) at iteration J is necessarily reached on $\phi_J(g)$, $g \ge 0$. More precisely it is attained over the range where $\phi_J(g)$ defines the maximum function. See Figure 3.

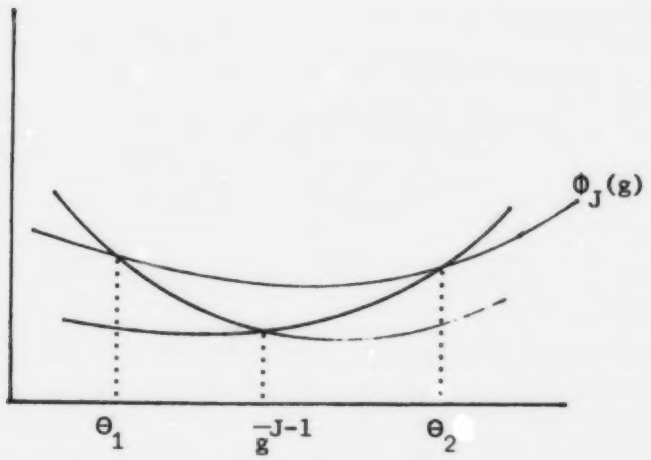

Figure 3
The Minimization Problem of Step 2

In light of the above proposition, the following solution procedure for problem (16)-(18) is proposed.

Step 1. Set $\theta_1 = 0$ and $\theta_2 = G$ (an upper bound of the travel demand g). If $J = 1$, go to Step 3.

Step 2. Compute

$$q_j = \frac{M^j - M^J}{\bar{u}^J - \bar{u}^j} \quad , \qquad j = 1,\ldots,J-1.$$

set

$$\theta_1 = \max \{\theta_1, (q_j \mid q_j < \bar{g}^{J-1})\}$$

and

$$\theta_2 = \min \{\theta_2, (q_j \mid q_j > \bar{g}^{J-1})\} .$$

Step 3. Solve the one-dimensional optimisation problem

$$\min \{\phi_J(g) , \ \theta_1 \le g \le \theta_2\}$$

for a $\bar{g}^J$ and $\bar{x}^J = \phi_J(\bar{g}^J)$. Terminate.

5. Computational Experience

In order to implement the algorithm described in Section 3 we have chosen to use a fixed-demand assignment method developed in [15]. The essence of this algorithm is the determination, for each O-D pair in turn, of an ϵ-equilibrium. The computations are repeated cyclically until no improvement is possible. For the sake of brevity, we give here only the basic computational steps for an O-D pair, which are as follows:

(Algorithm for fixed demand assignment)

Step 1: Compute travel times u_k for all paths $k \in K(i)$, where $K(i)$ is the set of paths available for the ith. O-D pair (if the initial solution is an "all-or-nothing" assignment then the set $K(i)$ contains initially one path).

Let s be the path with the smallest travel cost

$$u_s = \min_k \{u_k\}$$

Step 2: For each path $k \neq s$, compute the difference (descent direction) d_k defined as follows

$$d_k = h_k (u_s - u_k) \quad , \quad k \in K(i) - \{s\}$$

and set

$$d_s = -\sum_{k \neq s} d_k \quad .$$

If $d_s \leq \epsilon$ go to Step 5

Step 3: Compute ζ^* that minimizes

$$\sum_{a \in A} \int_0^{v_a + \zeta L_a} f_a(v)\, dv$$

$$\text{subject to } 0 \le \zeta \le \min\left\{ \frac{-h_k}{d_k} \,\middle|\, d_k < 0 \right\}$$

where v_a is total flow on arc a and

$$L_a = \sum_k \delta_{ak}\, d_k$$

and δ_{ak} is defined as in (6)

Step 4: Revise the path flows as follows

$$h_k = h_k + \zeta^* d_k \quad , \quad k \in K(i)$$

Return to Step 1

Step 5: If there is a path q with travel cost $u_q < u_s$ then revise the available set of paths $K(i)$ to include path q and return to Step 1; otherwise terminate

In Step 5 a shortest path calculation is required to determine u_q. The program that implements the above algorithm, uses Dijsktra's algorithm [6]. Labeled, nonpermanent nodes are stored as a HEAP. The determination of ζ^* in Step 3 is performed with the Golden Section Method [22].

As all the paths that carry flow are stored together with the corresponding flows, diminishing or augmenting the demand for any particular O-D pair is easily achieved. Furthermore, if the demand decreases, one does not have to perform the shortest path calculation in Step 5 of the above procedure.

Several test problems of differing size have been created with data for the street network, of the City of Hull, Canada. All problems are based on the same network of 155 nodes and 376 one-way links. We have varied the number of O-D pairs in order to have problems of increasing difficulty. Test problems are identified by the symbol ODn, where n is equal to the number of O-D pairs in the problem. The link travel costs are given by the travel time function suggested in BPR traffic assignment manual [17], which has the form:

$$f_a = t_a \left[1.0 + .15\left(\frac{v_a}{\ell_a}\right)^4\right]$$

where:

t_a = the free flow travel time

ℓ_a = the practical capacity of the link.

The inverse of the demand function $W_i(g_i)$ is a linear function of the form:

$$W_i(g_i) = b_i - a_i g_i \; ; \quad b_i, a_i > 0$$

We generate such functions by choosing the constants b_i and a_i in the following manner. We obtain first the fixed demand assignment using the original O-D demand matrix. Then b_i and a_i are obtained by fitting a line through the set of points (u_1, y_1) , (u_2, y_2) ; these points are randomly generated in the intervals given below.

$$\bar{u}_i^o \leq u_1 \leq \frac{3}{2}\bar{u}_i^o \quad , \quad \frac{1}{2}\bar{g}_i^o \leq y_1 \leq \bar{g}_i^o$$

$$\frac{1}{2}\bar{u}_i^o \leq u_2 \leq \bar{u}_i^o \quad , \quad \bar{g}_i^o \leq y_2 \leq \frac{3}{2}\bar{g}_i^o$$

The ϵ-equilibrium flow pattern in our program is characterized by the following conditions:

(i) ϵ-equilibrium with respect to supply function - all utilized paths have less than 5% difference in travel time

(ii) ϵ-equilibrium with respect to the demand function - the travel time computed from the inverse demand function, with the current demand, and the travel time of the shortest path have less than 5% difference.

The program was coded in Fortran and run on a CDC CYBER 74 with the optimizing FTN compiler. The experimental results are summarized in Table 1. These results may be summarized by the observation that obtaining the assignment with elastic demands only requires 20 to 25% more computational effort than the fixed demand assignment, and the running times are reasonably small.

These preliminary results are very encouraging although they are restricted to the functions used and the size of the network tested in our problems. However, one can reasonably conclude that if the demands used in the initial solution is not radically different from the equilibrium demands then the procedure seems to be quite efficient. This may be explained in part by the fact that a small departure from an equilibrium flow pattern, which occurs when a particular demand is adjusted, is efficiently corrected by the adopted traffic assignment algorithm. This is not a surprising result, because the fixed demand

traffic assignment algorithm has been developed within a mathematical programming framework and correcting a departure from an equilibrium pattern is a simple postoptimization procedure. In addition, Generalized Benders Decomposition seems to be very efficient for this problem. In all our tests, the largest number of constraints (17) that we required, was 4.

Thus, we conclude that, in practice, one should not hesitate to tackle problems with elastic demands whenever demand functions can be reasonably estimated.

	PROBLEM IDENTIFICATION	OD83	OD121	OD171	OD250	OD337	OD421
INITIAL EQUILIBRIUM FLOW	VALUE OF OBJ. FUNCTION ($x-10^6$)	.724517	.935304	1.193510	1.620354	1.758622	1.995031
	TOTAL TRIPS ASSIGNED ($x10^6$)	.126401	.141858	.154803	.167355	.175951	.181377
	DURATION (CPU SECS.)	5.92	10.74	17.89	26.81	34.54	43.42
FINAL EQUILIBRIUM FLOW	VALUE OF OBJ. FUNCTION ($x-10^6$)	.737116	.950099	1.208918	1.640018	1.783666	2.018451
	TOTAL TRIPS ASSIGNED ($x10^6$)	.127576	.143709	.152080	.168056	.177920	.184030
	HIGHEST O-D TRIPS CHANGE (%)	42.96	44.31	46.27	48.00	42.58	56.41
	TOTAL DURATION (CPU SECS.)	10.28	14.07	21.31	33.31	43.16	54.13

Table 1

Experimental results

APPENDIX

1. Generalized Benders Decomposition

Consider the problem

$$\underset{x,y}{\text{Min}} \quad f(x,y)$$

$$\text{Subject to} \quad G(x,y) = 0, \; x \in X, \; y \in Y$$

Generalized Benders Decomposition allows the solution of such problems by partitioning into a problem in variables x only and a problem in variables y only that must be solved several times. The algorithm, in a form suitable for our purpose, is as follows:

Step 1. Find a $\bar{y} \in Y$ and set $J = 1$.

Solve the problem

$$\underset{x \in X}{\text{Min}} \; f(x,\bar{y}) \quad \text{subject to} \quad G(x,\bar{y}) = 0$$

obtain an optimal (or near-optimal) solution and the corresponding multiplier vector $\bar{u}^J$. Select the convergence tolerance parameter $\varepsilon > 0$ and put $UBD = f(\bar{x}^J, \bar{y})$ Form the function

$$L^*(y,u) = \inf_{x \in X} \{f(x,y) + u^t G(x,y)\}, \; y \in Y, \; u \text{ unrestricted.}$$

Step 2. Solve the problem

$$\underset{y_o, y}{\text{Min}} \; y_o \quad \text{subject to} \quad y_o \geq L^*(y,\bar{u}^j), \; j = 1,\ldots,J$$

by an applicable algorithm and obtain $\bar{y}_o^J$, $\bar{y}^J$

If $\bar{y}_o^J \geq UBD - \varepsilon$ terminate, otherwise set $J = J + 1$

Step 3. Solve the problem

$$\min_{x \in X} f(x, \bar{y}^{J-1}) \text{ subject to } G(x, \bar{y}^{J-1}) = 0$$

and obtain new $\bar{u}^J$, $\bar{x}^J$; put UBD = min {UBD, $f(\bar{x}^J, \bar{y}^{J-1})$}

If $\bar{y}_o^{J-1} > \text{UBD} - \varepsilon$ terminate, otherwise go to Step 2.

2. Computation of functions $L^*(y,u)$ for our problem.

At each iteration a new constraint $L^*(g,\bar{u}^J)$ is added to the optimisation problem in Step 2 of the algorithm; $L^*(g,\bar{u}^J)$ was defined as

$$(23) \qquad L^*(g,\bar{u}^J) = \inf_{h_k \geq 0} \{ F(h,g) + \bar{u}^J (g - \sum_k h_k) \}, \quad g \geq 0,$$

where $$F(h,g) = \sum_a \int_o^{v_a} f_a(v)dv - \int_o^g w(y)dy,$$

and $$v_a = \sum_k \delta_{ak} h_k.$$

Since $F(h,g)$ is linearly separable in h and g, we may write

$$(24) \qquad L^*(g,\bar{u}^J) = \inf_{h_k \geq 0} \{ \sum_a \int_o^{v_a} f_a(v)dv - \bar{u}^J \sum_k h_k \}$$

$$+ \bar{u}^J (g - \int_o^g w(y)dy), \quad g \geq 0.$$

Recalling that $\bar{u}^J$ is the optimal multiplier obtained in the optimisation of the fixed-demand traffic assignment problem of Step 1, it follows from duality theory that

$$(25) \qquad \inf_{h_k \geq 0} \{ \sum_a \int_o^{v_a} f_a(v)dv - \bar{u}^J \sum_k h_k \}$$

$$= F(\bar{h}^J, \bar{g}^{J-1}) + \int_o^{\bar{g}^{J-1}} w(y)dy - \bar{u}^J \bar{g}^{J-1}.$$

$$= M^J$$

Thus, $L^*(g,\bar{u}^{J})$ is obtained with little effort, once the assignment problem of Step 1 is solved.

REFERENCES

[1] Beckman, M.J., McGuire, C.B., Winsten, C.B., "Studies in the Economics of Transportation", Yale University Press, New Haven, Conn., 1956.

[2] Bruynooghe, M., Gibert, A., Sakarovitch, M., "Une Méthode d'Affectation du Trafic", in Fourth Symposium on the Theory of Traffic Flow (W. Lentzback and P. Baron, eds.), Strassenban and Strassenverhelv-Stechnik, Bonn, 1969.

[3] Dafermos, S.C. and Sparrow, F.T., "The Traffic Assignment Problem for a General Network", Journal of Research, National Bureau of Standards-B, Vol. 73B, No. 2, 1969.

[4] Dafermos, S.C., "An Extended Traffic Assignment Model with Applications to Two-Way Traffic", Transportation Science, Vol. 5, No. 4, Nov. 1971.

[5] Dafermos, S.C., "The Traffic Assignment Problem for Multi-Class User Transportation Networks", Transportation Science, Vol. 6, No. 1, Feb. 1972.

[6] Dijsktra, E.W., "A Note on Two Problems in Connexion with Graphs", Numerische Mathematik, Vol. 1, 1959.

[7] Geoffrion, A.M., "Generalized Benders Decomposition", Journal of Optimization Theory and Applications, Vol. 10, No. 4, 1972.

[8] Gibert, A., "A Method for the Traffic Assignment Problem when Demand is Elastic", LBS-TNT-85, Transport Network Theory Unit, London Business School, London, August 1968.

[9] Kulash, D., "A Transportation Equilibrium Model", Urban Institute Paper 708-45, Washington, D.C., September 1971.

[10] Leventhal, T.L., Nemhauser, G.L. and Trotter, L.E., "A Column Generation Algorithm for Optimal Traffic Assignment", Transportation Science, Vol. 7, No. 2, 1973.

[11] Mangasarian, O.L., "Nonlinear Programming", McGraw-Hill, 1969.

[12] Martin, B.V. and Manheim, M.L., "A Research Program for Comparison of Traffic Assignment Techniques", Highway Research Record 88, 1965.

[13] Murchland, J.D., "Road Network Traffic Distribution in Equilibrium", paper for the Tagung Weber "Mathematische Methoden in den Wirtchafts Wissenschaften", Mathematisches Forschungsinstitut, Oberwolfach, October 1969.

[14] Netter, M. "Equilibrium and Marginal Cost Pricing on a Road Network with Several Flow Types", in Traffic Flow and Transportation, American Elseview, New York 1972.

[15] Nguyen, S., "A Mathematical Programming Approach to Equilibrium Methods of Traffic Assignment with Fixed Demands", publication # 138, Département d'informatique, Université de Montréal, août 1973.

[16] Ruiter, Earl R., "The Prediction of Network Equilibrium - The State of the Art", Transportation Systems Division, Department of Civil Engineering, Massachusetts Institute of Technology, June 1973.

[17] U.S. Department of Commerce, Bureau of Public Roads, Traffic Assignment Manual, Washington, D.C., 1964.

[18] Wigan, M.R., "Benefit Assessment for Network Traffic Models and Application to Road Pricing", Road Research Laboratories Report LT 417, Crowthorne, Berkshire 1971.

[19] Wigan, M.R. and Banford, T.J.G., "A Perturbative Model for Congested and Overloaded Transportation Networks", Road Research Laboratory Report LR 411, Crowthorne, Berkshire 1971.

[20] Wilkie, D.F. and Stefanek, .R.G., "Precise Determination of Equilibrium in Travel Forecasting Problems Using Numerical Optimization Techniques", Highway Research Record 369, 1971.

[21] Wardrop, J.G., "Some Theoretical Aspects of Road Traffic Research", Proc, Inst. Civil Engineers, Part II, Volume 1, 1952, pp. 325-378.

[22] Zangwill, W.I., "Nonlinear Programming: A Unified Approach", Prentice-Hall, Englewood Cliffs, N.J., pp. 121-122, 1969.